REMARKS

Claims 1-19 were presented for examination, and claims 1-19 stand rejected. In the current amendment, claims 16-18 have been amended. No new matter has been introduced. Upon entry of the current amendment, claims 1-19 will be presently pending in this application, of which claims 1, 14, and 19 are independent. Applicants submit that pending claims 1-19 are in condition for allowance.

The following comments address all stated grounds of rejection. The Applicants urge the Examiner to pass the claims to allowance in view of the remarks set forth below.

Claim Amendments

Claims 16-18 have been amended to clarify and more fully appreciate the Applicants' claimed invention. Support for the amended claims can be found on page 5, lines 7-26; page 9, lines 10-22; Figures 5, 6 and 12; and throughout the remainder of the specification. No new matter has been introduced. Applicants submit that the presently pending claims are in condition for allowance.

Claim Rejections Under 35 U.S.C. §102

I. Claims 1-2 and 8-19 Stand Rejected Under 35 U.S.C §102(e) As Being Anticipated By Lin

Claims 1-2 and 8-19 stand rejected under 35 U.S.C. §102(e) as being anticipated by Lin (U.S. Patent No. 6,631,452) ("Lin"). Applicants respectfully traverse this rejection.

A. Summary of Claimed Invention

Overflow and underflow conditions in a microprocessor trigger costly software traps that perform spill and fill operations on register windows. An overflow condition occurs when an

instruction requires a register window to be added and all of the registers are currently used. In response to the overflow condition, a trap occurs and the trap handler performs a spill operation to spill the contents of one of the registers into storage. An underflow conditions occurs when a current register does not contain contents required by an instruction, and the contents of the register window must be transferred from storage to a register. In response to the underflow condition, a trap occurs and the trap handler performs a fill operation to fill in the contents of a register with contents from storage. As such, spill and fill operations occur in traps responsive to overflow and underflow conditions from executing instructions.

The claimed invention is directed towards a microprocessor that comprises a mechanism to avoid an overflow and underflow condition from an upcoming instruction, and thereby, avoiding the costly trap that executes a spill and fill operation. The microprocessor comprises a detector for detecting an imminent register window overflow or underflow condition, and an instruction generator responsive to the detector for generating an instruction to avoid a trap responsive to the condition that is detected as imminent. That is, the instruction generator generates an instruction in response to the detector detecting an overflow or underflow condition from an upcoming instruction. The instruction manipulates storage to avoid the detected condition, and therefore the trap, when the upcoming instruction is executed. As such, the microprocessor does not execute the costly spill or fill operation of the trap.

B. Summary of Lin

Lin is directed towards a register management system that speculatively performs spill and fill operations in response to determining if there is available bandwidth in the memory channel. Lin describes a register stack engine (RSE) for performing register management and

handling mandatory and speculative spill/fill operations. The RSE manages data transfers between storage and register files when there is available bandwidth in the memory channel and when there is not a mandatory spill or fill operation from a pending instruction to execute. A mandatory spill or fill operation is caused by a trap trigged by an overflow or underflow condition from a pending instruction. If the RSE detects a mandatory spill or fill operation caused by a trap of an overflow or underflow condition, the RSE does not avoid the trap but executes the spill or fill operation of the trap (see Lin, Fig. 5). If the RSE is not performing any mandatory operations and there is available bandwidth, the RSE will perform speculative spill and fill operations (see Lin, Fig. 5). These spill and fill operations are in response to detecting available bandwidth, and not in response to an overflow or underflow condition. They are speculative because an instruction may require allocation of the speculatively spilled or filled register before the instruction associated with the spilled or filled data is re-activated (see Lin, column 8, lines 13-16). By its speculative nature, Lin reduces but does not avoid mandatory spill and fill operations which may stall processor execution (see Lin, column 5, lines 46-49; column 7, line 66 to column 8 line 16; column 8, lines 43-44 and 63-64, and column 10 lines 57-60).

C. Independent Claims 1 and 14 Patentably Distinguished Over Lin

Independent claims 1 and 14 are directed towards a microprocessor comprising a detector for detecting an imminent register window overflow or underflow condition, and an instruction generator *responsive to the detector* for generating an instruction to *avoid a trap from the condition that is detected as imminent*. That is, the detector detects that a register window overflow or underflow condition and resulting trap will occur from an upcoming instruction. In response to this detection, the generator generates an instruction to manipulate storage to avoid

the occurrence of the overflow or underflow condition and the resulting trap. In this manner, the microprocessor does not execute the spill or fill operations of the trap that would have occurred responsive to the detected condition from an upcoming instruction.

Lin does not disclose a mechanism for *generating an instruction* to *avoid a trap* from a register window overflow or underflow condition *in response to detecting the register window overflow or underflow condition.* In the case of detecting a mandatory spill or fill operation from an overflow or underflow condition, the RSE of Lin executes the mandatory operation (see Lin, column 8, lines 43-44, and Fig. 5). The spill or fill operation is mandatory because the overflow or underflow condition was detected and the trap triggered, without avoidance, to execute the mandatory operation. In contrast to the claimed invention, Lin does not generate an instruction to avoid the trap from the overflow or underflow condition. In response to detecting the overflow or underflow condition, Lin executes the trap's mandatory spill or fill operation.

In the case of speculative operations, Lin executes speculative spill or fills when it detects there is available bandwidth and when there is not a mandatory operation to execute (see Lin, Fig. 5). In contrast to the claim invention, Lin does not perform speculative spill or fill operations *in response to detecting a register window overflow or underflow condition.* Instead, Lin performs these speculative operations in response to detecting bandwidth availability. At the time of executing a speculative spill, there is not an instruction requiring a register to be saved to storage to make space available for the instruction, i.e., an overflow condition. Similarly, at the time of executing a speculative fill, there is not an instruction requiring a register to be restored from storage, i.e., an underflow condition. As such, the spill or fill operations are speculative in that they are not executed *in response to an imminent underflow or overflow condition.* Furthermore, even after a speculative spill or fill operation, a pending instruction may still cause

an overflow or underflow condition and therefore, a mandatory spill or fill operation to occur (see Lin, column 8, lines 13-16).

For the aforementioned reasons, Lin fails to disclose a mechanism for *generating an instruction* to *avoid a trap* from a register window overflow or underflow condition *in response to detecting the imminent condition.* Therefore, Applicants submit that claims 1 and 14 are patentable and in condition for allowance. Claims 2 and 8-13 depend on and incorporate the patentable subject matter of independent claim 1. Claim 15 depends on and incorporates the patentable subject matter of independent claim 14. Accordingly, Applicants request the Examiner to withdraw the rejection of claims 1-2 and 8-15 under 35 U.S.C. §102.

D. Amended Independent Claim 16 Patentably Distinguished over Lin

Independent claim 16, as amended, is directed to a method in a microprocessor to determine an imminent register window overflow or underflow condition, and, *in response to determining the condition*, manipulate the storage of register window contents to *avoid a trap responsive to the determined condition.* That is, in response to determining that a register window overflow or underflow condition will occur from an upcoming instruction, the method manipulates the contents of register windows in storage to avoid the occurrence of the trap from the determined condition. By this method, the claimed invention does not execute the spill or fill operation of the trap responsive to the register window overflow or underflow condition

Lin does not disclose a method for manipulating, *in response to determining that a register window overflow or underflow condition is imminent,* storage to *avoid a trap responsive to the determined condition.* As discussed above, when Lin determines there is an overflow or underflow condition from a pending instruction, a trap is triggered and mandatory spill or fill

operations are executed. As such, in response to detecting the condition, Lin does not manipulate storage to avoid the trap, but executes the spill or fill operation from the trap. When there are not any mandatory operations to execute, Lin performs speculative fill or spill operations in response to detecting the availability of bandwidth. In contrast to the claimed invention, Lin does not perform these speculative operations in response to any detected imminent overflow or underflow condition. As such, Lin performs spill or fill operations not to avoid a trap from a detected overflow or underflow condition, but to reduce the number of traps and resulting mandatory spill or fill operations that may speculatively occur later on.

For the aforementioned reasons, Lin fails to disclose a method in a microprocessor to determine an imminent register window overflow or underflow condition, and, *in response to determining the condition*, manipulate the storage of register window contents to *avoid a trap responsive to the determined condition*. Therefore, Applicants submit that amended claim 16 is patentable and in condition for allowance. Claims 17-19 depend on and incorporate the patentable subject matter of independent claim 16, as amended. Accordingly, Applicants request the Examiner to withdraw the rejection of claim 16-19 under 35 U.S.C. §102.

Claim Rejections Under 35 USC §103

II. Claims Rejected Under 35 U.S.C §103 As Unpatentable Over Lin

Dependent claims 3-7 are rejected under 35 U.S.C §103(a) as being unpatentable over Lin. Applicants respectfully traverse this rejection and contend that Lin fails to detract from the patentability of claims 3-7.

A. Non-obviousness of Claims Dependent from Patentable Independent Claim 1

As discussed above, Lin fails to disclose the patentable limitations of claim 1. Claims 3-7 depend on and incorporate all the patentable limitations of independent claim 1. Furthermore, Lin fails to teach or suggest each and every claim limitation of dependent claims 3-7. Independent claim 1 is directed towards a microprocessor that contains a detector for detecting an imminent register window overflow or underflow condition, and a generator *responsive to the detector* for generating an instruction *to avoid a trap responsive to the condition detected as imminent.*

Lin does not teach or suggest a microprocessor comprising a generator *responsive to a detector* for generating an instruction *to avoid a trap responsive to the condition detected as imminent.* In the case of detected underflow or overflow conditions, Lin seeks to perform the mandatory spill or fill operations of the traps triggered under such conditions. Lin does not discuss generating instructions to *avoid the trap in response to detecting the imminent condition.* In order to reduce overflow or underflow conditions that may speculatively be detected later on, Lin seeks to perform speculative fill and spill operations when processor bandwidth is available. Furthermore, Lin performs such speculative operations when there are not any underflow or overflow conditions detected, e.g., mandatory spill or fill operation. Lin is focused on more effective use of registers when memory bandwidth is available, and when mandatory spill or fill operations are not executing (see Lin, Fig. 5). As such, Lin does not teach or suggest a generator *responsive to the detector* for generating an instruction *to avoid a trap responsive to the register window overflow or underflow condition detected as imminent.*

For the aforementioned reasons, Applicants submit that Lin does not detract from the patentability of claim 1. As such, claims 3-7 dependent from claim 1 are patentable and in

condition for allowance. Therefore, Applicants respectfully request the withdrawal of the Examiner's rejection of claims 3-7 under 35 U.S.C. §103.

B. Non-obviousness of Dependent Claims 3-7

In addition to the patentable limitations of independent claim 1, dependent claims 3-7 are directed towards a microprocessor further comprising a cache for caching instructions for introduction into an execution stage and wherein the microprocessor's detector examines the instructions in the cache to determine if a register window overflow or underflow condition is imminent. In the Office action, the Examiner admits that Lin does not teach the limitations recited in these dependent claims. In the rejection of these dependent claims, the Examiner cites Lin to suggest one ordinarily skilled in the art might modify Lin to look further up the instruction path into the instruction cache to see what instructions are coming up. Applicants contend that one ordinarily skilled in the art at the time invention was made would not find a motivation or suggestion in Lin to modify Lin as the Examiner suggests.

Lin does not teach or suggest looking further up the instruction path into the instruction cache to see what instructions are coming up. The RSE of Lin is a state machine that monitors the processor and maintains the register stack based on the currently pending instruction (see Lin, column 8, lines 59-60). It is focused on the current state of the register stack engine and the currently pending instruction of the processor, if any. When there is not a pending instruction, the RSE checks for available bandwidth to perform speculative spill and fill operations. As such, the RSE state machine is not prospective in nature, and does not look further up the instruction path for future instructions. Moreover, if Lin did teach or suggest looking further up the instruction path, which it does not, Lin would not need to perform any speculative spill or fill

operations. The spill or fill operations are speculative because Lin does not check further up the instruction path. As such, Lin does not teach or suggest looking at future instructions. Therefore, one ordinarily skilled in the art would not find a motivation or suggestion in Lin to modify the state engine of Lin to look further up the instruction path for future instructions.

For the aforementioned reasons, Applicants submit that Lin does not detract from the patentability of claims 3-7. Accordingly, Applicants respectfully request the withdrawal of the Examiner's rejection of claims 3-7 under 35 U.S.C. §103.

Conclusion

In light of the aforementioned arguments, Applicants contend that each of the Examiners rejections has been adequately addressed and the pending application is in condition for allowance.

Should the Examiner feel that a telephone conference with Applicants' attorney would expedite prosecution of this Application, the Examiner is urged to contact the Applicants' attorney at (617) 227-7400.

Respectfully submitted,
LAHIVE & COCKFIELD, LLP

Dated: November 9, 2004

By Christopher J. McKenna
Christopher J. McKenna
Registration No.: 53,302
Attorney For Applicants

Lahive & Cockfield, LLP
28 State Street
Boston, Massachusetts 02109
(617) 227-7400
(617) 742-4214 (Fax).